The value of C depends on the packing factor, p, and on the length of string N. It is given by

$$C = \sum_{i=0}^{(N/p)-1} 2^{(p-1)i} \quad \text{with } p > 2$$

for p-RB1 numbers, and by

$$C = \sum_{i=1}^{N/p} 2^{(p-1)i} \quad \text{with } p > 2$$

for p-RB2 numbers.

Detection of the zero

In redundant binary representations of signed numbers, the number $(0)_{10}$ has many representations, consequently it is not easily detectable. On the contrary, the number $(-1)_{10}$ can be detected in two logical steps, as proved for the RB1 representation in Ref. 13. This fact can be generalized as: any redundant binary representation of the number $(-1)_{10}$ obtains the canonical form of $(-1)_{10}$ if the following rules are applied only on the *nr* pair

RB1 or p-RB1	RB2 or p-RB2
$01 \rightarrow 10$	$01 \rightarrow 01$
$10 \rightarrow 10$	$10 \rightarrow 01$

The validity of these rules can be proved following the same outline used in Ref. 13 in the case of the RB1 representation.

Efficiency of p-RB numbers

The efficiency of packed binary redundant number representations can be written as[13]

$$E = \frac{R_B}{R_R} = \frac{p-1}{p} \tag{9}$$

where R_B is the number of bits needed in the binary number system and R_R is the number of bits needed in a redundant number representation to write the same value. Equation (9) shows that efficiency increases when the packing factor, p, increases. In fact, for p-RB numbers one can write

$$\lim_{p\to\infty} E = \lim_{p\to\infty} \frac{p-1}{p} = 1$$

The behaviour of E versus p is shown in Table 5. Notice that, for $p \rightarrow \infty$, the p-RB2 representation coincides with the natural binary number system

$$\lim_{p\to\infty} \sum_{i=0}^{N-1} a_i 2^{i-\lfloor i/p \rfloor} = \sum_{i=0}^{N-1} a_i 2^i$$

Concluding remarks

Redundant number representations are very suitable for optical computing, but they have efficiencies lower than the binary number system. This fact implies more expensive hardware. The p-RB number representations, introduced in the previous sections, have greater efficiency than the RB representation even though they maintain the same advantages: they naturally fit the 2's complement binary number system and they permit one to build a fast and inherently parallel arithmetic.

Table 5. Efficiencies of the p-RB number representation versus the packing factor p

p-RB	E
2-RB	0.500
3-RB	0.666
4-RB	0.750
5-RB	0.800
6-RB	0.833
7-RB	0.857
8-RB	0.875
9-RB	0.888
10-RB	0.900
11-RB	0.909
12-RB	0.916
...	...
binary	1.000

The algebraic sum can be performed in constant time ($p + 4$ elemental logic steps) for any efficiency value, using symbolic substitution with very small truth tables. For these reasons, p-RB numbers can be considered as a good trade-off between speed and cost of optical computing devices.

References

1 **Guest, C.C., Gaylord, T.K.** Truth-table look-up optical processing utilizing binary and residue arithmetic, *Appl Opt*, **19** (1980) 1201–1207

2 **Brenner, K.-H.** New implementation of symbolic substitution logic, *Appl Opt*, **25** (1986) 3061–3064

3 **Brenner, K.-H., Huang, A., Streibl, N.** Digital optical computing with symbolic substitution, *Appl Opt*, **25** (1986) 3054–3060

4 **Mirsalehi, M.M., Gaylord, T.K.** Truth-table look-up parallel data processing using an optical content addressable memory, *Appl Opt*, **25** (1986) 2277–2283

5 **Vuillemin, J.** A very fast multiplication algorithm for VLSI implementation, *INTEGRATION, the VLSI Journal*, **1** (1983) 39–52

6 **Takagi, N, Yasuura, H., Yajima, S.** High-speed VLSI multiplication algorithm with a redundant binary addition tree, *IEEE Trans Comput*, **C-34** (1985) 789–796

7 **Avizienis, A.** Signed-digit number representations for fast parallel arithmetic, *IRE Trans Electron Comput* **EC10** (1961) 389–400

8 **Bocker, R.P., Drake, B.L., Lasher, M.E., Henderson, T.B.** Modified-signed addition and subtraction using optical symbolic substitution, *Appl Opt*, **25** (1986) 2456–2457

9 **Li, Y., Eichmann, G.** Conditional symbolic modified signed-digit arithmetic using optical content-addressable memory logic elements, *Appl Opt*, **26** (1987) 2328–2333

10 **Cherri, A.K., Karim, M.A.** Modified-signed digit arithmetic using an efficient symbolic substitution logic, *Appl Opt*, **27** (1988) 3824–3827

11 **Alam, M.S., Awwal, A.A.S., Karim, M.A.** Digital optical processing based on higher-order modified signed-digit symbolic substitution, *Appl Opt*, **31** (1992) 2419–2425

12 **Awwal, A.A.S.** Recoded signed-digit binary addition subtraction using opto-electronic symbolic substitution, *Appl Opt*, **31** (1992) 3205–3208

13 **De Biase, G.A., Massini, A.** Redundant binary number representation for an inherently parallel arithmetic on optical computers, *Appl Opt*, **32** (1993) 659–664

14 **Huang, K.-S., Jenkins, B.K., Sawchuck, A.A.** Binary image algebra and optical cellular logic processor design, *Computer Vision, Graphics, Image Process*, **45** (1989) 295–345

15 **Huang, K.-S., Jenkins, B.K., Sawchuck, A.A.** Image algebra representation of parallel optical binary arithmetic, *Appl Opt*, **28** (1989) 1263–1278

16 **De Biase, G.A., Massini, A.** Parallel optical arithmetic on images by a redundant binary number representation, *Appl Opt*, **29** (1990) 1587–1589